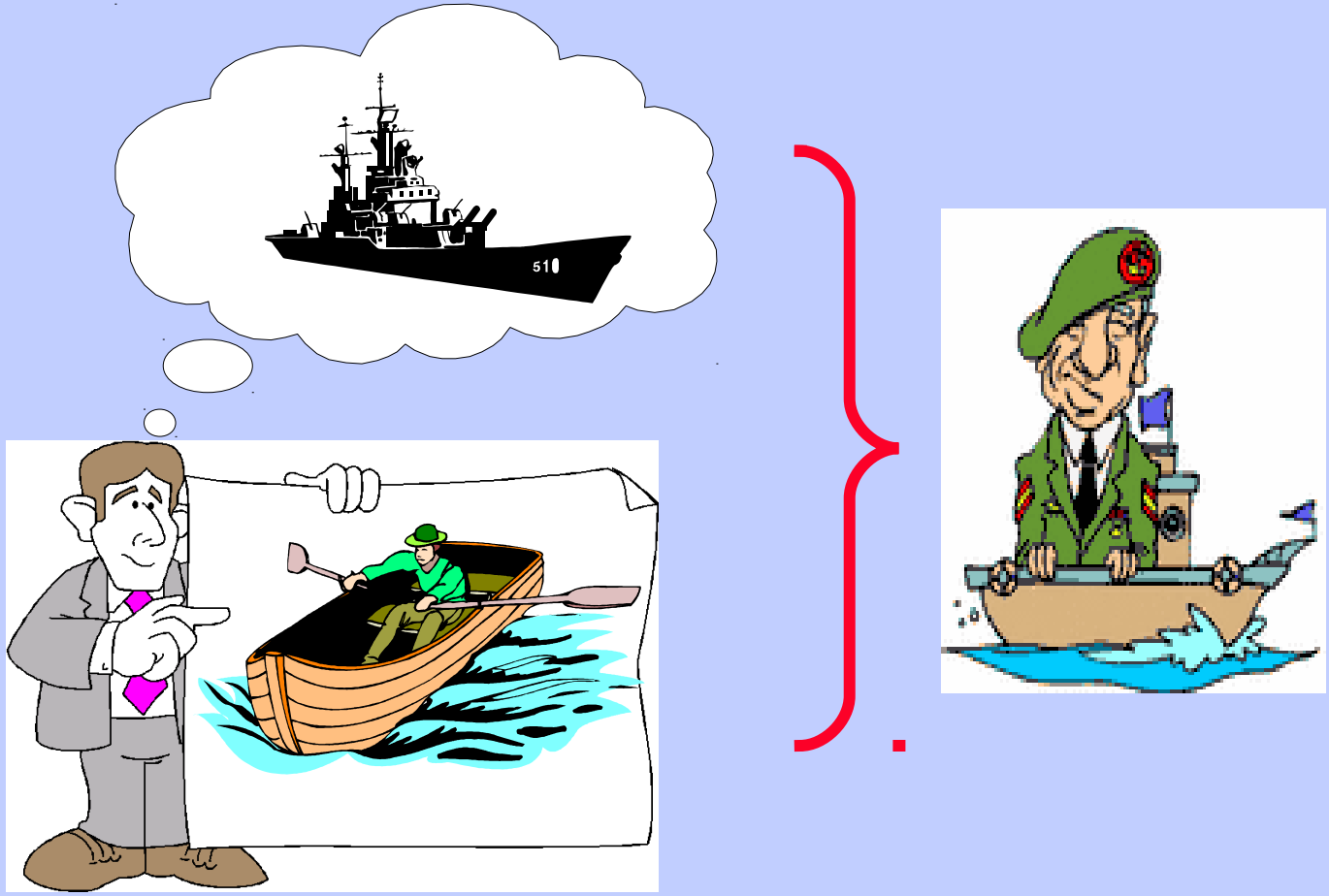
SPAWAR
Systems Center
San Diego

Systems and Software Requirements
Management and *Development*



SPAWAR
Systems Center
San Diego

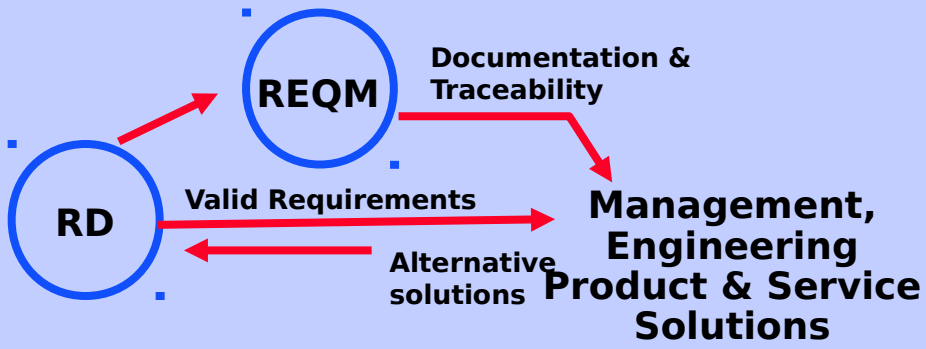
Frequently Asked Questions

1. What do we mean by managing requirements? How does that relate to developing requirements?
2. What is a requirement?
3. Why are requirements a problem?
4. What is a requirements spiral?
5. What is the requirements process and who are the stakeholders?
6. What are some effective requirements gathering techniques?
7. How do you manage the evolution and explosion of requirements?
8. How do you write primitive and effective requirements?
9. How can I get started?
10. What other references are available for requirements topics?

SPAWAR
Systems Center
San Diego

1. Requirements Management vs Development

- **Requirements Management (REQM)**
  - **Focuses on basic fundamental activities of documentation and *traceability***
  - **Builds dialogue and *accountability***
  - ***Manages changes* to requirements and inconsistencies between requirements, work products and plans**



- **Requirements Development (RD)**
  - **Focuses on *discovery* of customer needs**
    - **Internal / external**
  - ***Transforms* the "voice of the customer" into product qualities or service functions and levels, and an understanding of what the customer will gain**
  - ***Validates* (based on evidence or sound footing) the requirements before going into production or servicing the customer**
  - **Usually comes first in the lifecycle**

SPAWAR
Systems Center
San Diego

Requirements Management

- To manage the requirements of the project's products and product components and to identify inconsistencies between those requirements and the project's plans and work products
- To “manage requirements” means to:
  - Document current requirements
    - Received or generated by the project and / or the organization
    - Including both technical and non-technical requirements (e.g., processes, standards)
  - ***Have a plan for requirements management activities***
  - Have a list of “approved requirements providers”
  - Ensure project team commits to the requirements
  - Ensure that the “approved requirements” are the basis of project plans / activities
  - Document requirements changes and rationale
  - Maintain ***bi-directional traceability*** between source requirements and all product and product-component requirements

SPAWAR
Systems Center
San Diego

Requirements Management (2)

- **Document the official channels, sources, and representatives *who are eligible to provide requirements* (To avoid requirements creep)**
- **Document who are eligible project *"requirements receivers"***
- **Reach a "shared understanding" *on the meaning of the requirements***
- **Bi-directional traceability of requirements means *traceability from the requirement source to its lower level requirements, its work product, and back.* It includes:**
  - **Intermediate and final work products**
  - **Changes in design documentation, test plans, and work tasks**
  - **Horizontal and vertical relationships, such as across interfaces**
  - **Requirements changes on the project plans, activities, and work products**
  - **Allocation of functions, objects, people, processes, and work products**

REQ
M

SPAWAR
Systems Center
San Diego

Requirements Development

- Builds on basic requirements management activities
- Its purpose is to ***produce and analyze*** customer, product and product-component requirements
- Details ***how*** to develop (come up with) requirements by:
  - ***Eliciting, analyzing, validating and communicating*** customer needs, expectations, and constraints to obtain customer requirements that constitute an understanding of what will satisfy stakeholders
  - Collecting and coordinating stakeholder needs
  - Developing the life-cycle requirements of the product
  - Establishing the customer requirements
  - Establishing the initial product and product-component requirements consistent with customer requirements
- ***Manages allocated requirements***

SPAWAR
Systems Center
San Diego

Requirements Development (2)

- Requires a process to understand, define, refine, and select the requirements at all levels.
  - ***Develop a plan for requirements development activities***
  - Analyze needs and requirements for each product life-cycle phase, including:
    - » Needs of stakeholders
    - » The operational environment
    - » Factors that reflect overall customer and end-user expectations and satisfaction, such as safety, security, and affordability
    - » Impact analysis on other allocated requirements
  - Develop an operational concept / view
  - Define the required functionality
- Requirements are allocated to products, product components, people, associated processes, or services (a change from CMM)
- Place designated work products of the requirements development process under appropriate levels of configuration management

RD

SPAWAR
Systems Center
San Diego

2. What Is A Requirement?

- **A requirement is a statement which identifies a capability, characteristic, or quality factor.**

3. Why Are Requirements A Problem?

The Standard Project Conundrum

People think in high-level needs – not the specifics.

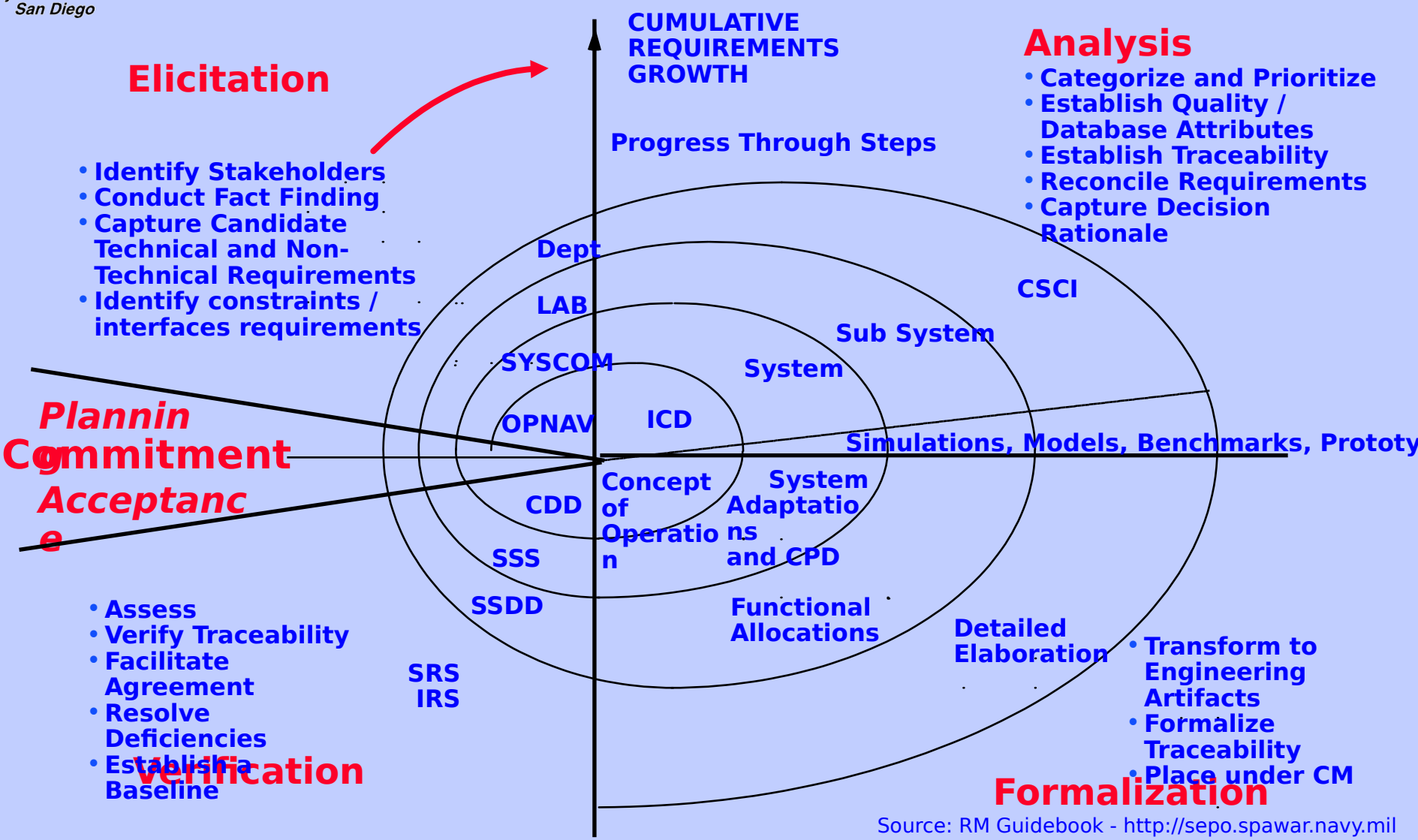
How do you transform:

"I want it to be fail-safe"
"I want it to be user-friendly"
"I want a miracle"

into working requirements that meet the project objectives?

4. The Requirements Spiral – A Risk Mitigator Crabwalk Through Managing and Developing Activit

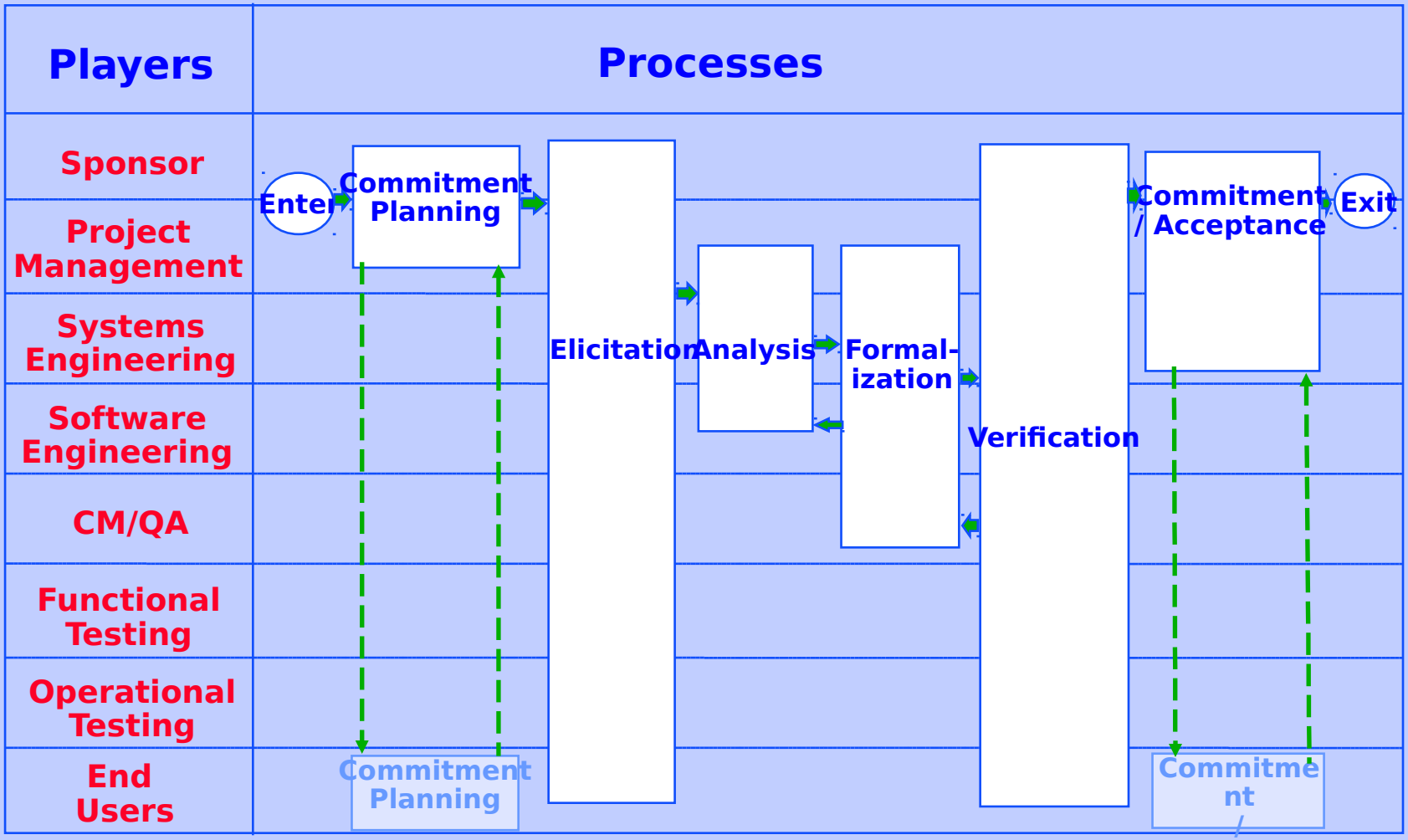
SPAWAR
Systems Center
San Diego



Source: RM Guidebook - http://sepo.spawar.navy.mil

Source: RM Guidebook - http://sepo.spawar.navy.mil/ under Requirements Management PA

5.1 The Stakeholders In The Requirements Process



For further information, see Additional Materials

SPAWAR
Systems Center
San Diego

Commitment / Planning

Participants: Sponsor, Project Management, End Users

Entry Criteria: The need for planning and agreement on the implementation of new or changed requirements

Inputs	Processing	Outputs
Mission Need Operational Scenario Change Requests Baselined Requirements Verification Reports	1. Define Acceptance Criteria for this evolution of the spiral 2. Identify Non-Technical Requirements 3. Develop/Update the Project Plan; to include cost/schedule impact. **4. Assess Project Risk** 5. Assign/Review Engineering Policy and Responsibilities 6. Review Project Plan 7. Review by Sponsor 8. Show Commitment to the Plan	Approved Plan Acceptance Criteria

Exit Criteria: Agreement among the stakeholders on scope of task, assignments, and schedules. Resources and funds allocate
An approved plan

SPAWAR
Systems Center
San Diego

Elicitation

Participants: Sponsor, Project Management, Systems Engineering, Software Engineering, QA, Functional Testing, Operational Testing, End Users

Entry Criteria: Agreement on the plan, program risks, resources and have been identified, responsibilities are allocated and understood

Inputs	Processing	Outputs
Approved Plan Mission need/change Derived Requirements	1. Identify Stakeholders 2. Conduct Fact Finding 3. Capture Candidate Technical Requirements 4. Capture Non-Technical Requirements 5. Identifying stakeholder constraints and interfaces	Candidate Technical Requirements Non-Technical Requirements Supporting Rationale

Exit Criteria: An understanding of the candidate technical requirements been obtained and supporting rationale documented

SPAWAR
Systems Center
San Diego

6. Proven Requirements Gathering Techniques

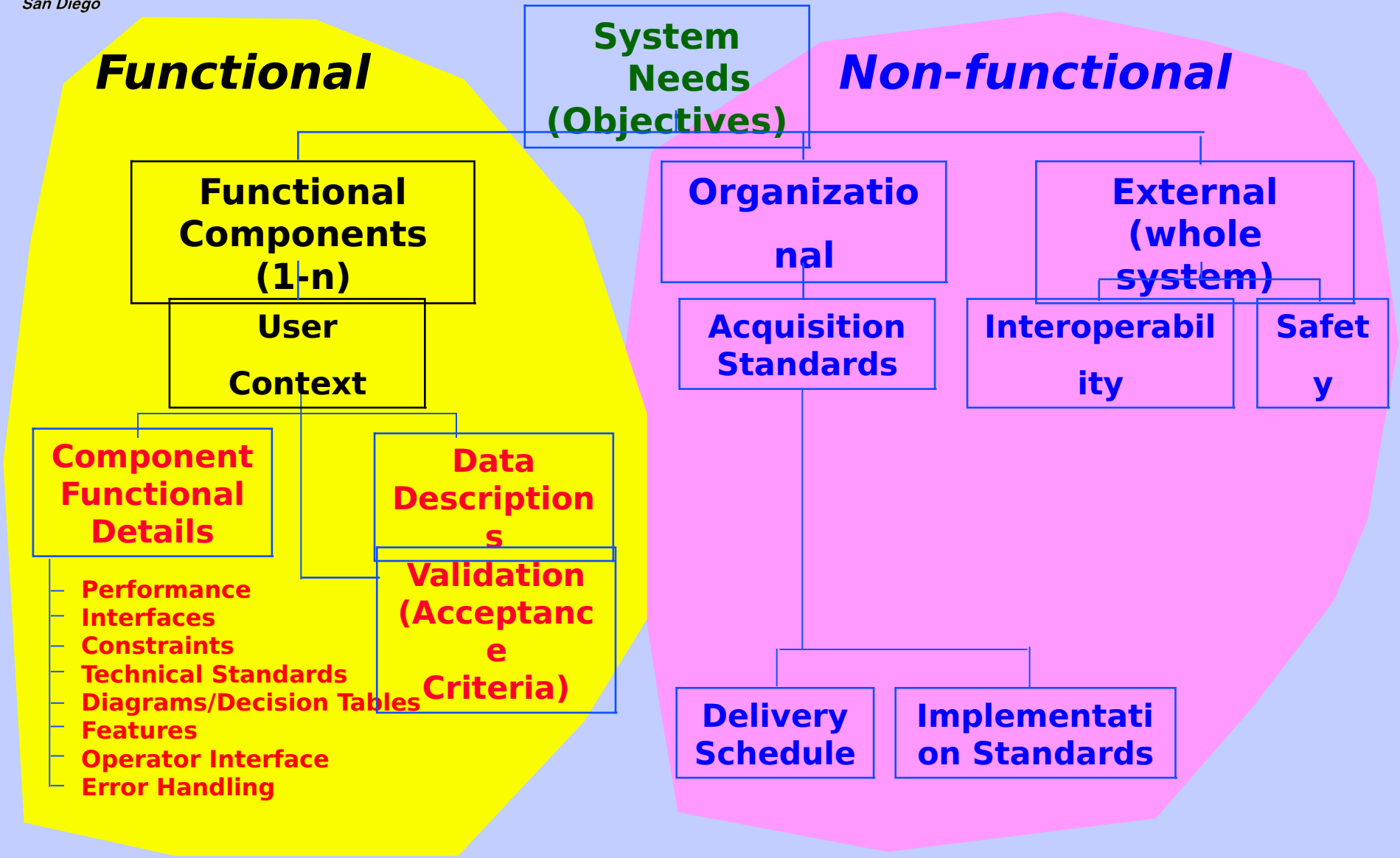
- **Interviews – What is the reason for solving this problem / implementing the requirement? What environment will product encounter?**
- **Document analysis – review business plans, white papers, requests for proposals, statements of work**
- **Requirements workshops – encourage brainstorming and consensus, are interactive and prioritize needs**
- **Use cases - picture of actions or scenarios a system performs**
- **Prototyping & story boarding – example / rough version / illustration of desired system to help facilitate the further definition of requirements**
- **Interface analysis and modeling – clarifies product scope and behavior, and interoperability among pieces**

Recommended Requirements Gathering Practices, CrossTalk, April 2002, STSC

http://www.stsc.hill.af.mil

SPAWAR
Systems Center
San Diego

Deriving the Specifics



SPAWAR
Systems Center
San Diego

Analysis

Participants: Project Management, Systems Engineering, Software Engineering

Entry Criteria: An understanding of the candidate technical requirem and supporting rationale

Inputs	Processing	Outputs
Candidate Technical Requirements Non-Technical Requirements Supporting Rationale	1. Categorize and Prioritize 2. Determine Quality Attributes 3. Establish Traceability 4. Reconcile Requirements 5. Capture Decision Rationale	Informal Requirements Initial Requirements Traceability

Exit Criteria: Informal requirements have been produced through assessment of the candidate requirements, requirements have been categorized, traceability established, and decision rationale has been captured and reconciled

SPAWAR
Systems Center
San Diego

Requirement Characteristics

Complete	**Unambiguous**
Correct	**Verifiable**
Feasible	**Consistent**
Necessary	**Modifiable**
Prioritized	**Traceable**

See handout entitled:
"Characteristics Of Good Requirements' Specifications "

SPAWAR
Systems Center
San Diego

Formalization

Participants: Project Management, Systems Engineering, Software Engineering, QA, CM

Entry Criteria: Informal requirements produced through the assessment of the candidate requirements are categorized, prioritized and placed in a database, traceability is establishment and decision rationale is reconciled

Inputs	Processing	Outputs
Informal Requirements Requirements Traceability	1. Transform Requirements to Formalized Engineering Artifacts 2. Formalize Traceability of Engineering Artifacts 3. Place Formalized Engineering Artifacts under Configuration Control	Formalized Requirements as engineering artifacts and traceability work products under CM.

Exit Criteria: The formal requirements/formal engineering artifacts and formal traceability work products are placed under CM control

SPAWAR
Systems Center
San Diego

Verification

Participants: Sponsor, Project Management, Systems Engineering, Software Engineering, QA, CM, Functional Testing, Operational Testing, End User

Entry Criteria: Requirements as formal engineering artifacts and formal traceability work products under CM

Inputs	Processing	Outputs
Formalized Requirements Formalized Eng Artifacts Formalized traceability work products	1. Perform Assessment 2. Verify Traceability 3. Resolve Deficiencies 4. Facilitate Agreement 5. Establish a Req. Baseline	Baselined Requirements Verification Report Anomaly Report

Exit Criteria: Formalized engineering artifacts have been reviewed, approved and placed under CM control. Any deficiency reports generated during the review cause a return to a previous activity for investigation

SPAWAR
Systems Center
San Diego

Commitment / Acceptance

Participants: Sponsor, Project Management, Senior Management, En

Entry Criteria: Baselined requirements and verification reports

Inputs	Processing	Outputs
Baselined Requirements Verification Report	1. Present Evidence of Baselined Requirements 2. Present Project Status 3. Formalize Commitment by All Stakeholders to Baselined Requirements 4. Obtain Approval to Proceed to the Next Process	Authenticated Requirements Approval to Proceed

Exit Criteria: The customer has approved the baselined requirements as meeting the acceptance criteria. Approval to proceed to the next evolution of the requirements management process is obtained

SPAWAR
Systems Center
San Diego

Managing the Evolution of Requirements

- Commit to a documented process to transform the statements of need to workable detailed requirements:
  - Use an Operational Concept Definition to initiate the transformation
  - Follow the process to inform, coordinate and educate all stakeholders
- Apply Configuration Management disciplines to manage defined requirements:
  - Use Configuration Control Boards, Interface Working Groups
  - Apply status accounting disciplines in tracking requirement status
  - Establish requirement baseline controls
- Maintain horizontal and vertical requirements traceability:
  - To design / specs, code / implementation / service, and test / verification
  - To interfaces
  - To parent and subordinate documents (i.e., system specification, plans, test cases, reports, validation)

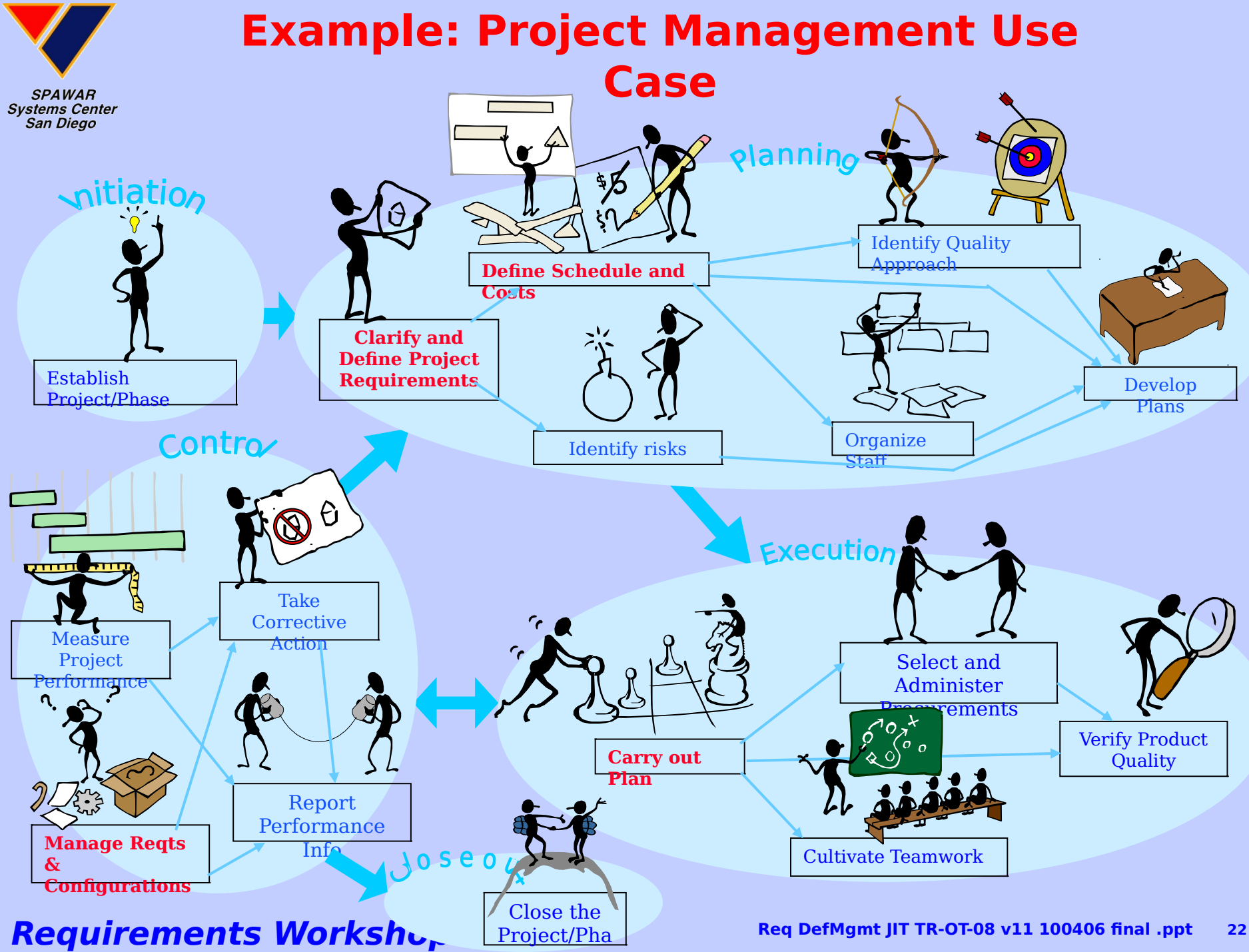
SPAWAR
Systems Center
San Diego

First Step in the Transformation - The Operational View

- Develops a common vocabulary
  - The basis for customer / developer data exchange
- Identifies all interfaces (including HCI)
- Builds an information baseline for growth
- Serves to identify and communicate assumptions
- Helps reveal risk areas
- Identifies needed core requirements
  - Functional and non-functional requirements
    - » ***Functional requirements*** are statements of the *services* that the system should provide
    - » ***Non-functional requirements*** are ***constraints*** on the services and functions offered by the system

Example: Project Management Use Case

SPAWAR Systems Center San Diego



Example: Operational View

SPAWAR Systems Center San Diego

Project identified
PM identified
SSC assets

INITIATION

2.1 Establish Project / Phase
a. Determine initial customer needs and requirements
b. Determine SSC needs
c. Become familiar with the SSC SD Strategic Plan
d. Determine mgt & engineering processes
e. Identify management and team responsibilities

Project Description
PM prepared
Constraints
Assumptions
Proposals
SOW

PLANNING

3.1 Clarify & Define Project Reqts.
a. Gather candidate reqts.
b. Analyze reqts
c. Formalize reqts
d. Review & clarify reqts and standards

3.2 Define Schedule & Costs
a. Estimate parameters
b. Document life cycle
c. Identify tasks, phases
d. Identify processes
e. Develop schedule
f. Estimate costs
g. Define resources

3.3 Identify Quality Approach
a. Define QA approach
b. Define CM approach
c. Identify methods to remove defects
d. Determine project measures

3.4 Organize Staff
a. Define org structure
b. Identify staff roles
c. Identify needed contracts
d. Determine stakeholder involvement
e. Identify training needs

3.5 Identify Risks
a. Document candidate risks
b. Determine risk avoidance
c. Document reduction/ contingency actions
d. Determine risk measures

3.6 Develop Plans
a. Document plans
b. Reconcile resources
c. Conduct peer reviews
d. Obtain plan approval & commitment

Revisions

Baselined Requirements
Project Plan
Schedule & Milestones

CONTROL

5.1 Measure Project Performance
a. Collect data
b. Analyze data, determine status
c. Monitor project and process quality

5.2 Manage Reqts & Configurations
a. Control project scope
b. Manage project and doc configurations

5.3 Take Corrective Action
a. Analyze issues
b. Take action
c. Implement risk plans

5.4 Report Perfor-mance Info
a. Report status
b. Communicate with stakeholders
c. Submit project measures/ work products

Project Status

Direction & Revisions

Completed products
Status measures

EXECUTION

4.1 Carry Out Plan
a. Design product or service solution
b. Develop product
c. Review for defects
d. Evaluate product against

4.2 Select and Administer Procurements
a. Acquire outside support
b. Administer outside support

4.3 Cultivate Teamwork
a. Acquire staff members
b. Build teamwork
c. Obtain training

4.4 Verify Product Quality
a. Implement QA
b. Report work results

CLOSEOUT

6.1 Close the Project / Phase
a. Deliver and support
b. Prepare de-staffing plan
c. Track/turn in equip. & hazardous matls
d. Conduct admin closure
e. Properly close out contracts

Delivered product
Returned equipment and recycle materials
Archived files

SPAWAR
Systems Center
San Diego

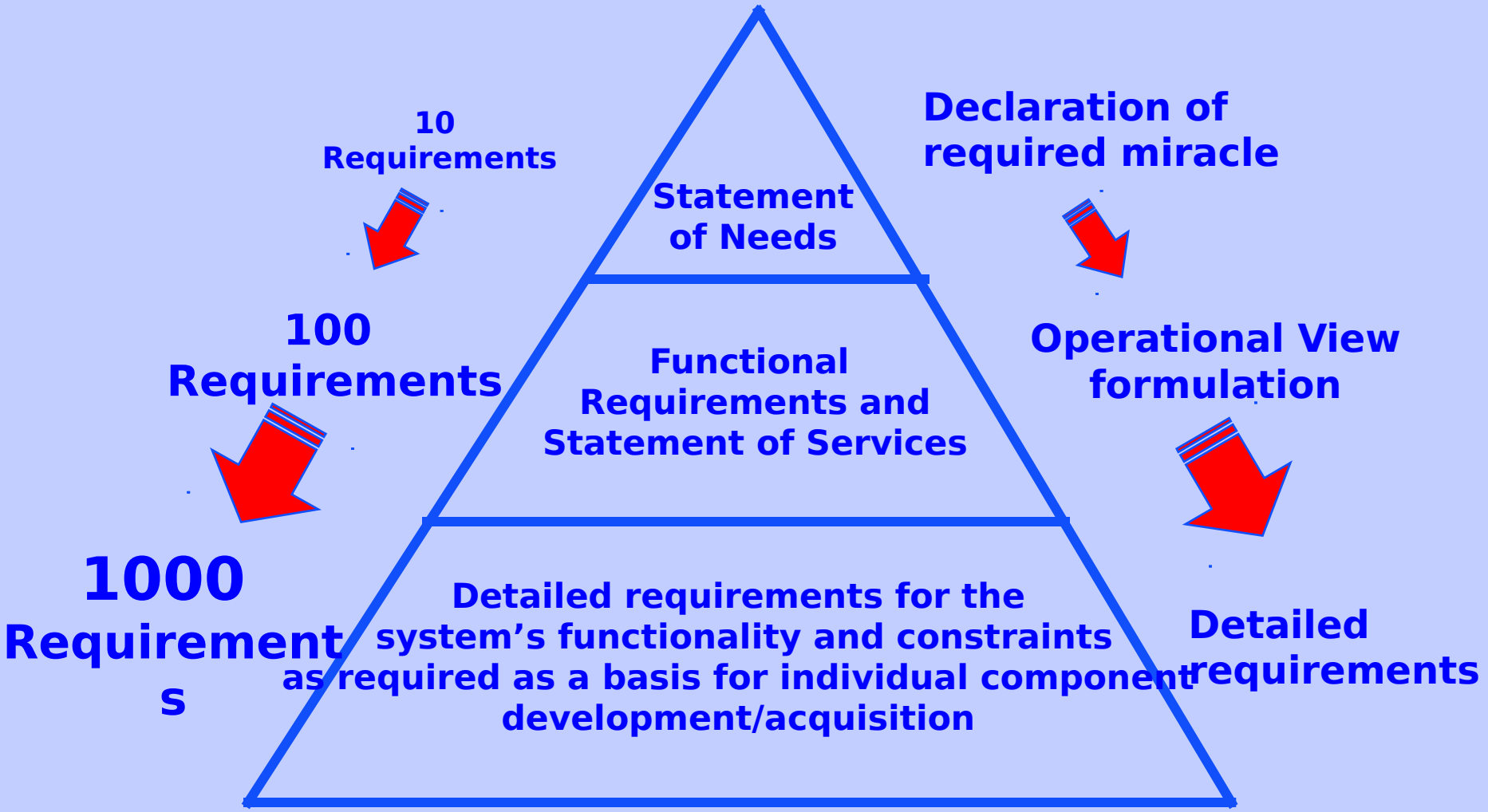
More on Managing Requirements

- Requirements must be stated in detail so as to be ***verifiable or testable.*** How can a requirement be met if it cannot be verified or validated?
- Requirements volatility within baselines must be controlled. Management of requirements change is essential for keeping a project on track. (e.g., Configuration Management)
- Requirements status must be tracked. ***Quantify*** the status of requirements as to the number:
  - Drafted
  - Completed
  - Baselined
  - Tested
  - Traceable to design or specs
  - Traceable to implementation (fabrication, code, etc.)
  - Traceable to verification (test, analysis, validation)
  - Traceable to parent and interfac
- Use a ***Requirements Review*** to communicate st requirements and resolve open issues
- ***Bring requiremen review sessions***
- Apply database te nage requirement

Automate!

SPAWAR
Systems Center
San Diego

Requirements Explosion



SPAWAR
Systems Center
San Diego

7. Managing the Explosion

- Commit to a documented process to ***transform the statements of need to workable detailed requirements:***
  - Use the operational view to guide the transformation
  - Apply industry-established functional standards
  - Follow the process to inform, coordinate and educate all stakeholders
- Separate the Human Computer Interface (HCI) requirements from the functional requirements. HCI is more dynamic and will require special attention as part of the Human Systems Integration (HSI).
- Apply ***configuration management*** disciplines to manage defined requirements:
  - Use Configuration Control Boards, Interface Working Groups
  - Apply status accounting disciplines in tracking requirement status
  - Establish requirement baseline controls
- Maintain requirements ***traceability*** relationships to:
  - Design, implementation, verification,
  - Parent documents (i.e., system specification)
- ***As requirements change, so must the traceability linkage***

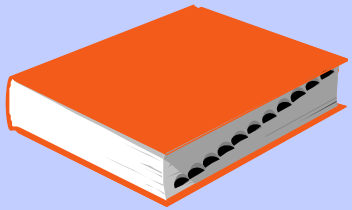
SPAWAR
Systems Center
San Diego

Uses of Database Technology on Automated Comm's Management System (ACMS)

- Requirements Identification and Engineering
- Requirements Traceability
- Audit Trails
- Compliance Checking
- Discrepancy Tracking
- Analysis and Design Support
- Configuration Management
- Documentation Support

SPAWAR
Systems Center
San Diego

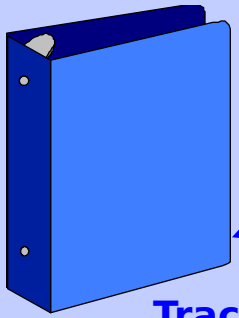
ACMS Document Support Concept



High Level Source Documents

Users

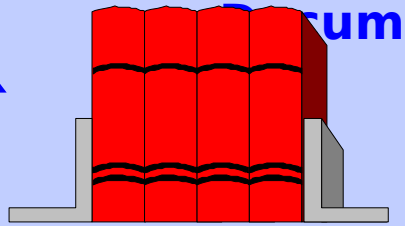
ACMS Databases



Report

- Traceability Matrix
- Discrepancy Listings
- Requirement Attribute Reports
- Status Reports
- Metrics Listings
- Configuration Control Log
- IV&V Log

Build Project Documents



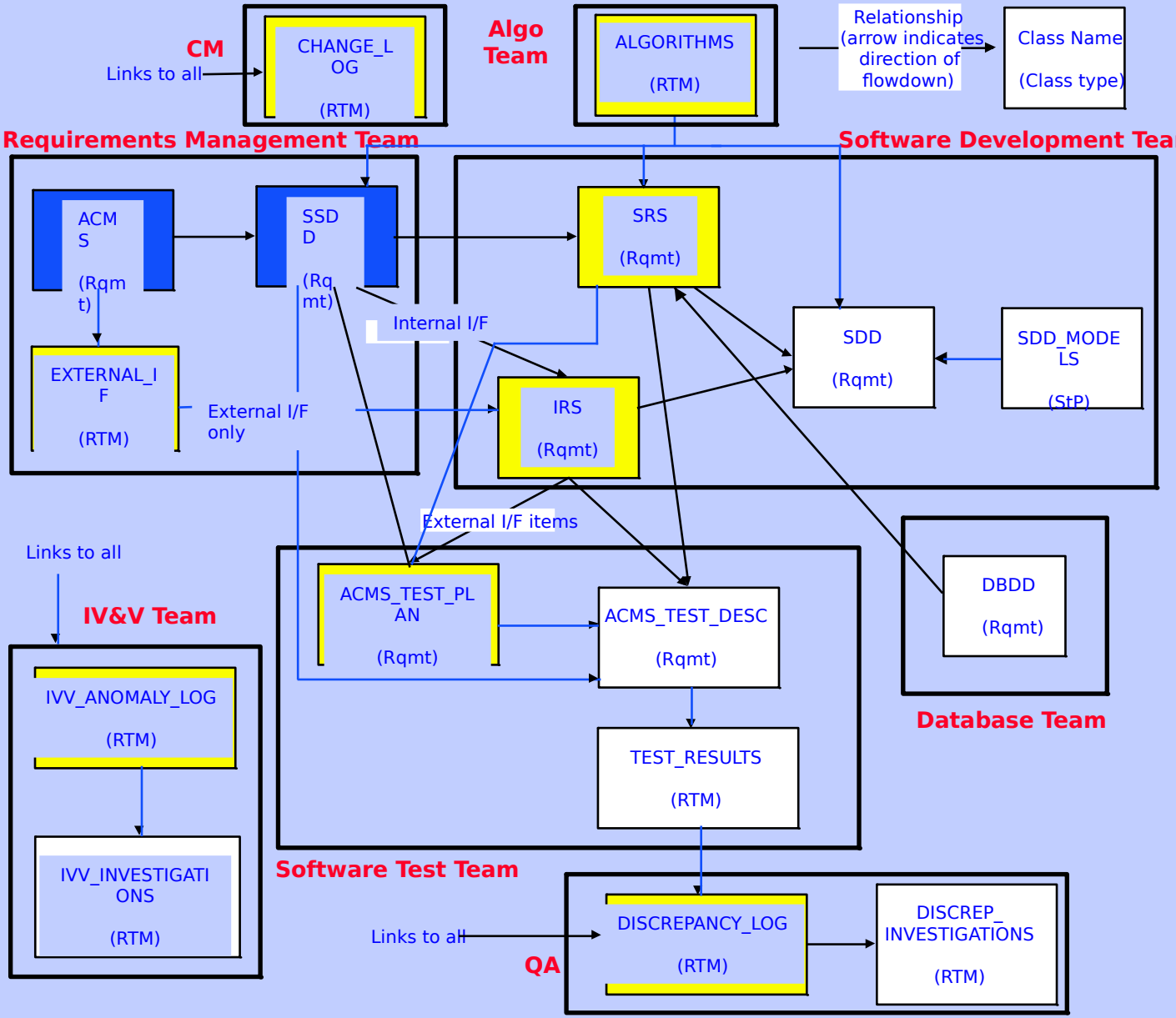
- SSDD
- SRS
- IRS
- DBDD
- Test Plan
- Test Descriptions

SPAWAR
Systems Center
San Diego

The Function of "Normalizing" Higher-Level Requirements

- Break requirements statements into discrete requirements
- Check for testability
- Check for closure on clarity
- Identify the source of the requirements
- Attribute the requirements
- Eliminate ambiguity
- Establish the requirements database

SPAWAR
Systems Center
San Diego



SPAWAR
Systems Center
San Diego

Basic RM Database Attributes

- Requirement Catalogue Numbers
- Parent Document Paragraph Numbers
- Requirement Text
- Comments
- Creation Date
- Last Modified Date
- Priority
- Status

- Revision Identification
- Test Plan Identification
- Test Methodology
- IV&V Intensity Level
- Design Derived (yes or no)

SPAWAR
Systems Center
San Diego

Example Database-Generated Build Schedule Repo

SSDD ID	CSCI	SSDD Function	Bl	In	SSDD Text
SS1000-a-	CPM	DISTRIBUTE_DATA	1	2	The ACMS shall use Milstar systems time on the ACMS to ACMS interface as specified in the ACMS to ACMS EXTIRS-B1-U-XXXX.
SS1000-b-	CPM	DISTRIBUTE_DATA	1	2	The ACMS shall use UTC on the ACMS to ACMS interface as specified in the ACMS to ACMS EXTIRS-B1-U-XXXX.
SS1160a	CPM	DISTRIBUTE_DATA	1	2	The ACMS shall interface to other ACMSs as specified in the ACMS to ACMS EXTIRS-B1-U-XXXX.
SS1160b	SOE	MANAGE_DATABASE	1	2	The ACMS shall interface to other ACMSs as specified in the ACMS to ACMS EXTIRS-B1-U-XXXX.
SS1160c	SOE	PROVIDE_SOE_SERVICES	1	2	The ACMS shall interface to other ACMSs as specified in the ACMS to ACMS EXTIRS-B1-U-XXXX.
SS2000-b-	CPM	DISTRIBUTE_DATA	1	1	The ACMS shall exchange communicatior planning data with other ACMSs as specified in the ACMS to ACMS EXTIRS-B1-U-XXXX.
SS2000-b-	CPM	DISTRIBUTE_DATA	1	2	The ACMS shall exchange communicatior planning data with other ACMSs as specified in the ACMS to ACMS EXTIRS-B1-U-XXXX.
SS2000-b-	CPM	DISTRIBUTE_DATA	1	3	The ACMS shall exchange communicatior planning data with other ACMSs as specified in the ACMS to ACMS EXTIRS-B1-U-XXXX.

SPAWAR
Systems Center
San Diego

Writing Good Requirements

- **Writing what people think they need is easy, *but mostly off the mark***
- **The hard work is:**
  - **Knowing what to write about**
  - **Determining appropriate values**
  - **Achieving balance between:**
    - » **Ensuring specificity (explicit conditions) and avoiding ambiguity**
    - » **Ensuring completeness and sufficiency (adequate detail to fulfill needs) without redundancy**

See handout entitled: "Tips For Writing Good Requirements"

Software Project Survival Guide, by Steve McConne

SPAWAR
Systems Center
San Diego

Writing Good Requirements (2)

- Understand who your audience is
  - Who is the user of the requirement?
  - Who is the recipient of the functionality?
  - Who verifies requirement?
  - Who validates requirement?
- Know the language and the business sector (e.g., aerospace, forces afloat);
  - Create primitive requirement statements; focus on attributes
  - Move up to simple complete sentences: single ideas per sentence are easier to verify and test
  - Decompose more complex sentences into simple single sentences
  - Elaborate with follow-on comments
- Rigid compliance with a format
- Develop and review “Use Cases” as ways of describing system behavior under various conditions
- Plan requirements activities and work products
- Manage changes to requirements

SPAWAR
Systems Center
San Diego

8. Writing Primitive Requirements

Progressive Process

- **Understand the attributes of the requirements statements**
  - **Characteristic (or attributes) identification**
  - **Relation**
  - **Value and units assignment**
- **Place in table or database for easier management and data validation**
- **Each gets own unique numbering**
- **Convert table to simple declarative sentences**
  - **Add items, verbs, sentence endings**
  - **Add connective phrases (if needed)**
  - **Add *only those words* which necessarily elaborate, clarify or amplify**
  - **Add punctuation**

Sentences are now suitable for inclusion in specification

SPAWAR
Systems Center
San Diego

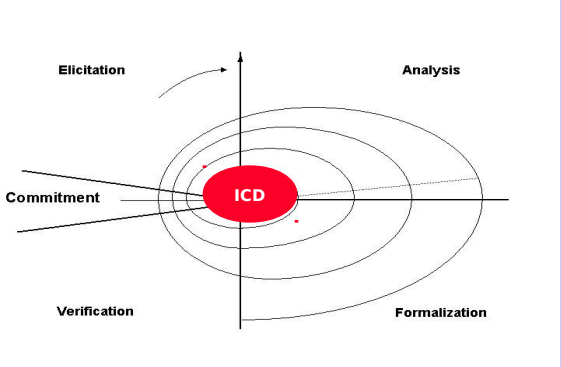
Writing Primitive Requirements: Examples

Controlled Attribute (What must be controlled?)	Relation =, >, < Comply with As defined in	Value & Units (Number or standard and units)
Weight of payload	less than or equal to	2400 pounds
Lifecycle	complies with	IEEE Std 12207
Screen refresh rate	equal to	20 frames / second

An order string of these three entities

SPAWAR
Systems Center
San Diego

For Example

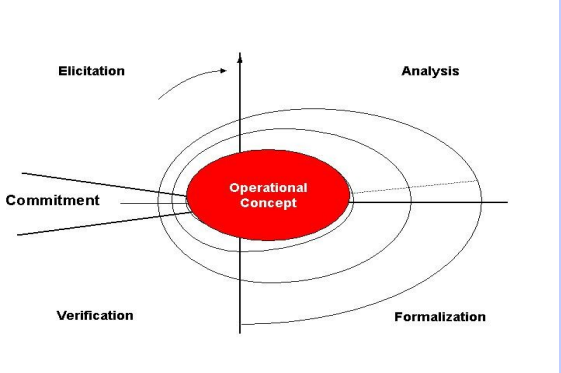


Human Resources has identified a need for a thermostat for the sight impaired.

THE STATEMENT OF NEED

SPAWAR
Systems Center
San Diego

For Example (2)



- I walk over to the thermostat and say something that gets thermostat's attention
  - It gives me verbal reply that it is listening
- I ask for "setting".
  - It tells me: set for cool, fan is on automatic, and the setting is 78 degrees F
  - It tells me that it is currently 79 degrees F
  - It asks me if I want to change the setting
- If I say no,
  - I'm done
- If I say yes,
  - It asks me if I want to change the temperature setting

THE OPERATIONAL VIEW

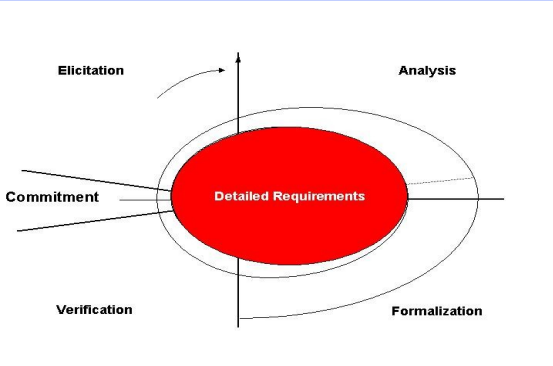
SPAWAR
Systems Center
San Diego

For Example (3)

Controlled Attribute (What must be controlled?)	Relation =, >,< Comply with As defined in	Value & Units (Number or standard and units)
device	complies with	UL safety standards
device	operates in	audio range of 20 to 50 decibels

SPAWAR
Systems Center
San Diego

For Example (4)



- **Technical Standards:**
  - **The device shall comply with UL standards for safety.**
- **Operator Interface:**
  - **The device shall understand the vocabulary in Table '*n*'**
- **Performance Requirements:**
  - **The device shall work at a range of 0 to 10 feet from audible direction source**
  - **The device shall work in the audio range of 20 to 50 decibels**

DETAILED REQUIREMENTS SPECIFICATION (i.e., SRS)

SPAWAR
Systems Center
San Diego

Requirements Documentation

- **Who will use the document?**
  - **Each role has a differing level of technical knowledge, subject matter expertise, and uses**
    - » **Whose job tasks are defined by this requirements document?**
    - » **Who needs to review or approve this document?**
    - » **Who is responsible for ensuring that this document and other project documentation meets your business, process and deliverable standards?**

Examples:

- **The project team uses this as the source of what the product / solution should provide, and uses the requirements as the linchpin for other work products**
- **Line managers use requirements documents to gain support for the product / solution, and to demonstrate the value of the system**
- **SME uses the requirements document to tell the user how the product / solution will meet the users needs, and solicits feedback**
- **Verifiers and QA use the requirements document to determine what the product / solution does to ensure that the design encompasses all the requirements, that all requirements are accounted for in verification, and that the testing proves the product / solution does what was intended**

Adapted from "Writing Requirements Documentation for Multiple Audiences", http://www.sticky

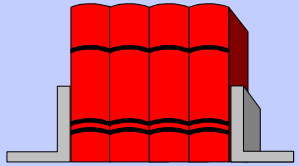
SPAWAR
Systems Center
San Diego

9. Activities to "Get Started"

- Review SEM Policy for managing system requirements
- Select accepted standards to ensure application of best practices and establish product requirements
- Plan for requirements activities (Contact SPI Agent To Help!)
- Locate, review, modify, adopt, and use desktop procedures for analyzing the system requirements and allocating them to hardware, software, and other system components
- Provide adequate resources and funding for managing the requirements (system level through allocated requirements)
- Evaluate and acquire tools to support the activities for developing, managing, and changing requirements
  - Does the tool fit my current process?
  - What risks will ***required process modifications*** engender?
- Train the engineering staff, and related team members to perform their requirements management activities
- Assign personnel with experience and expertise in the application and in systems / software engineering to manage the requirements (system level through allocated requirements)
- Ensure that requirements are placed in a database and establish traceability among and between the requirements, work products, and interfaces

Tall Order

SPAWAR
Systems Center
San Diego



10. References

- DoD Instruction 5000.1, The Defense Acquisition System, DoD, May 2003
- DoD Instruction 5000.2, Operation of the Defense Acquisition System, DoD, May 2003
- CMU/SEI-2006-TR-08, Capability Maturity Model Integration for Development Version 1.2 (CMMI DEV V1.2), SEI, August 2006
- ISO/IEC15288 System Engineering, System Life Cycle Processes, International Standards Organization, November 2002
- SIGSOFT Volume 15 No. 5, Spiral Development Strategy, ACM, October 1990.
- IEEE/EIA 12207, Software Life Cycle Processes, IEEE, March 1998
- ANSI/IEEE Std 830-1984, IEEE Guide to Software Requirements Specifications, IEEE, 1984
- NAVAIR Requirements Management Working Group, Requirements Management Guidebook, NAVAIR, Sept 1998
- SSC San Diego, A Description of the SSC San Diego Software Process Assets (SPA), SEPO, October 2001. See http://sepo.spawar.navy.mil/
- Cockburn, A., Writing Effective Use Cases, Addison-Wesley, 2001

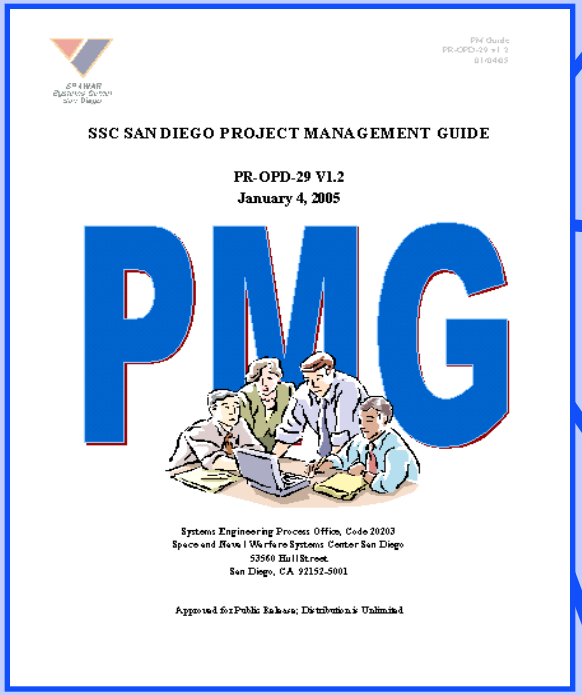
SPAWAR
Systems Center
San Diego

Additional Material

- Additional Material & References
  - PMG Guidance of Requirements Activities
  - Operational View
  - Top Level Requirements
  - Hierarchy of Detailed Requirements
  - Allocating Requirements to Components
  - Requirements Development Process (Expert Mode)
  - Requirements Management Process (Expert Mode)
  - Characteristics of Good Requirements Specifications
  - Tips For Writing Good Requirements

SPAWAR
Systems Center
San Diego

PMG Guidance on Requirements Activities


SSC SAN DIEGO PROJECT MANAGEMENT GUIDE
PR-OPD-29 V1.2
January 4, 2005
PMG
Systems Engineering Process Office, Code 20203
Space and Naval Warfare Systems Center San Diego
53560 Hull Street
San Diego, CA 92152-5001
Approved for Public Release; Distribution is Unlimited

3.1 Clarify and Define Project Rqmts
- *a. Gather candidate requirements*
- *b. Analyze requirements*
- *c. Formalize requirements*
- *d. Review and clarify requirements and standards*

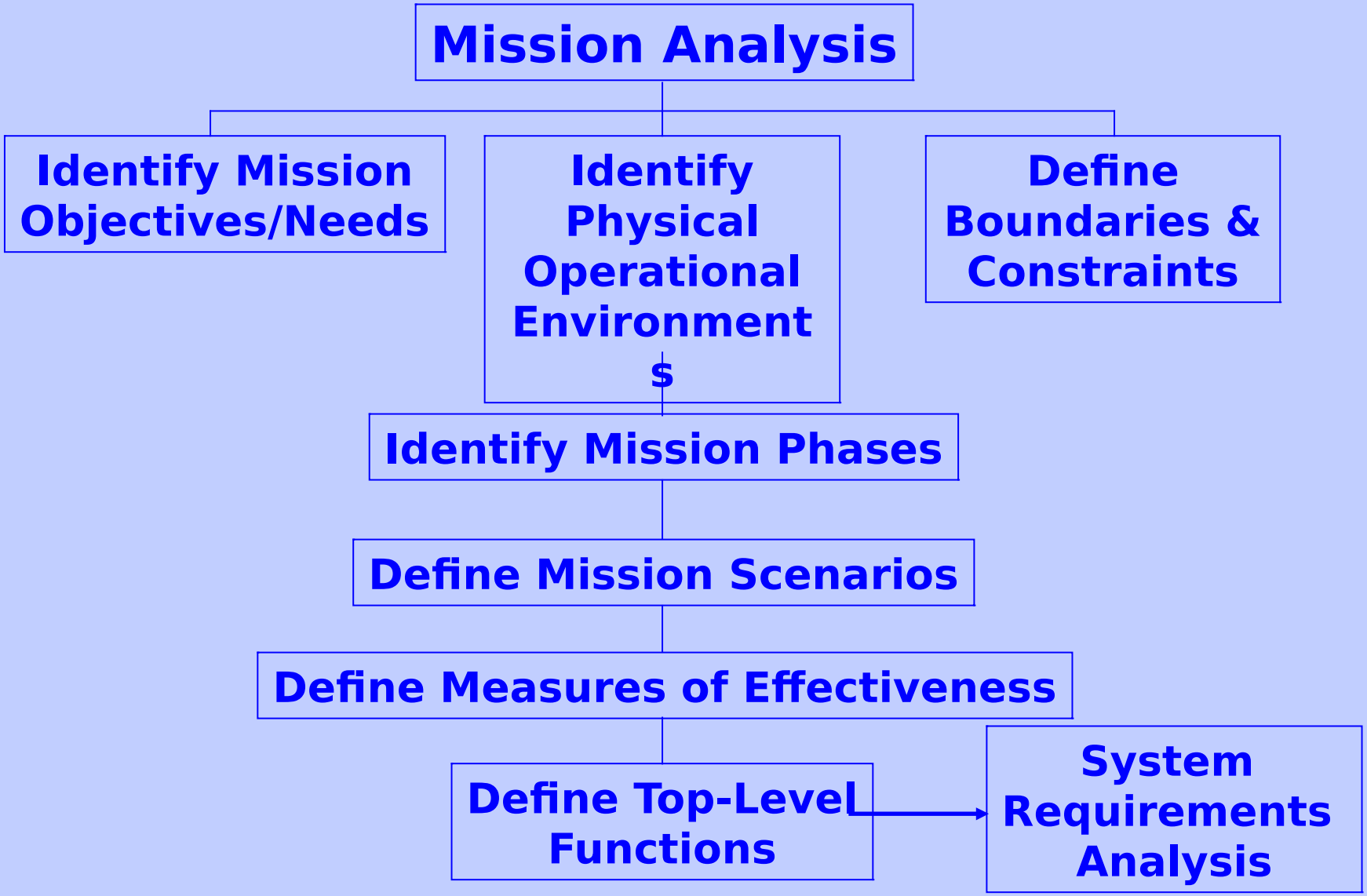
3.2 Define Schedule and Cost
- *a. Estimate the parameters*
- *b. Document life cycle*
- *c. Identify tasks , phases*
- *d. Identify the processes*

4.1 Carry out Plan
- a. Design the product or service
- b. Develop the product
- c. Review for defects
- d. Evaluate the product against requirements

5.2 Manage Requirements and Configurations
- *a. Control project scope*
- *b. Manage product and document configurations*

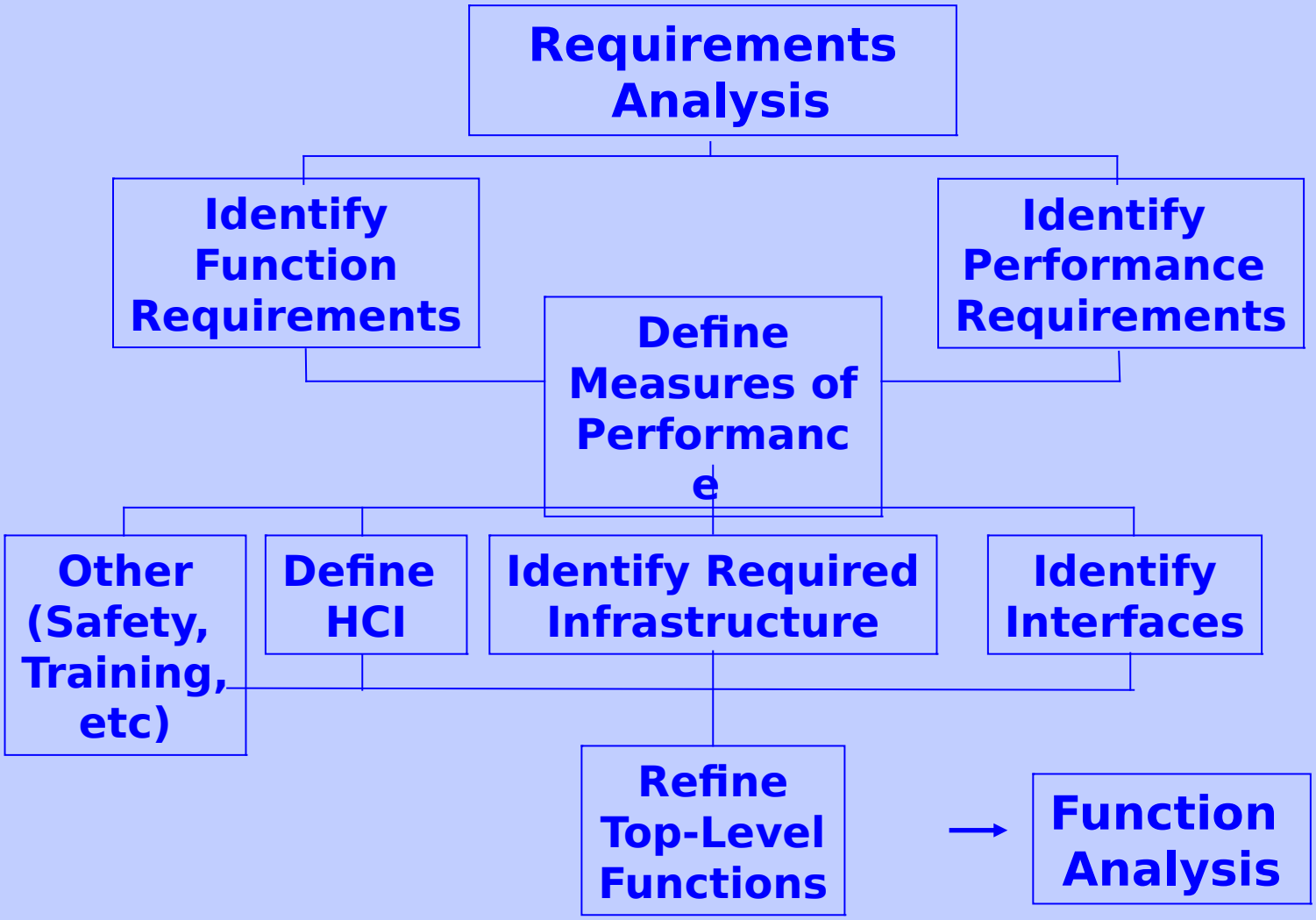
SPAWAR
Systems Center
San Diego

Define Operational View



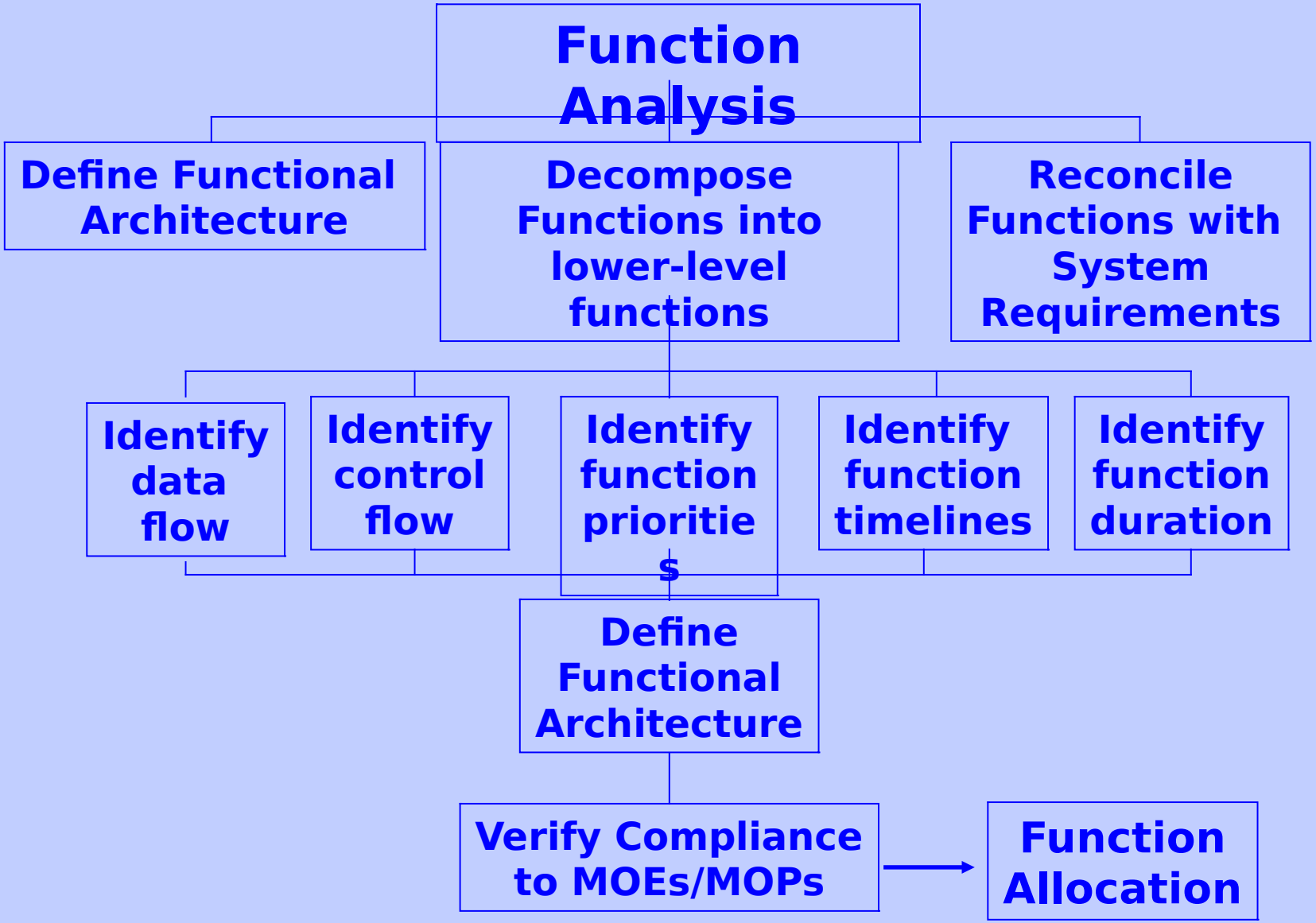
SPAWAR
Systems Center
San Diego

Develop Top-Level Requirements



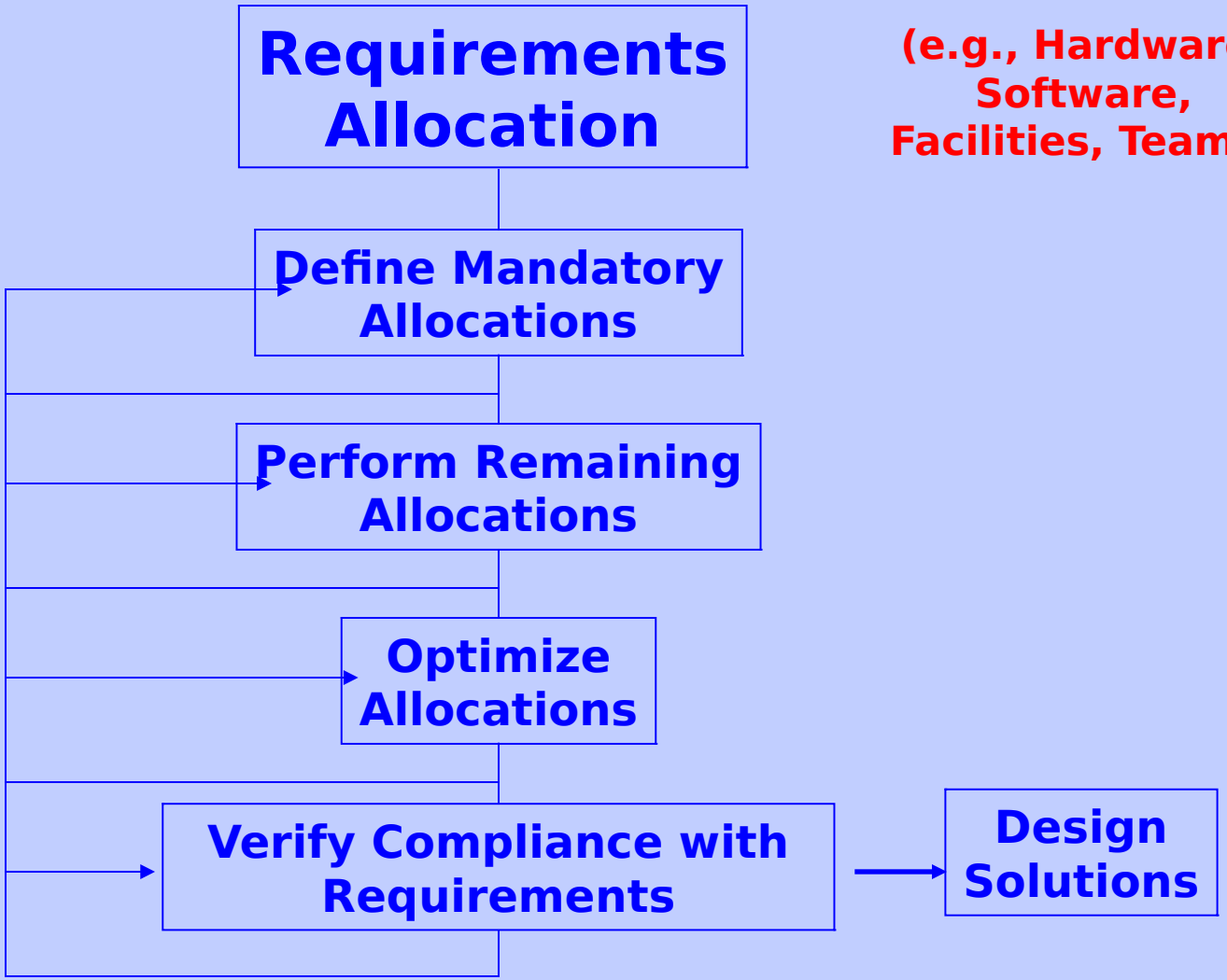
SPAWAR
Systems Center
San Diego

Develop Hierarchy of Detailed Requirements



SPAWAR
Systems Center
San Diego

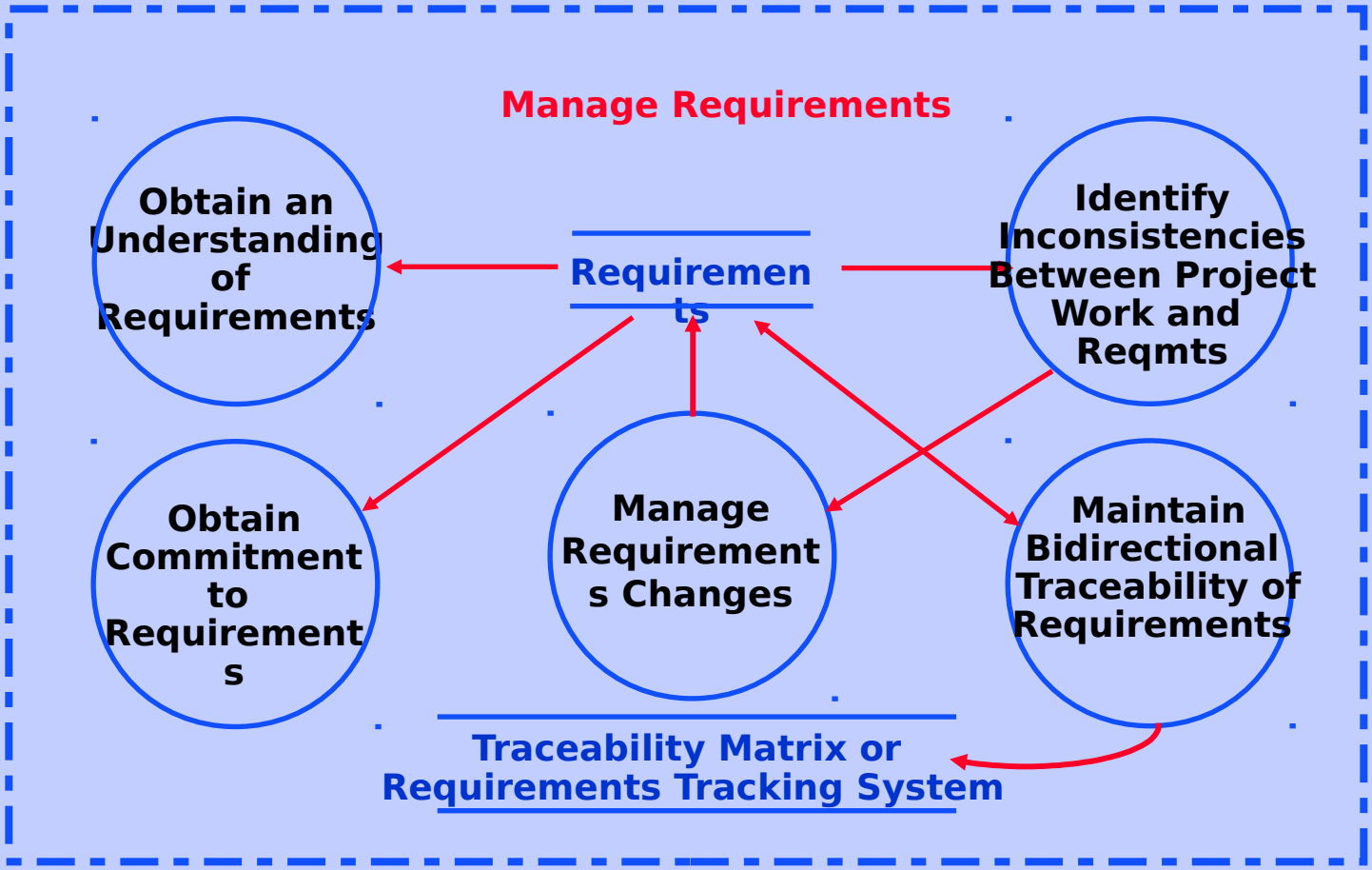
Allocating Requirements to Components



(e.g., Hardware, Software, Facilities, Teams)

Return

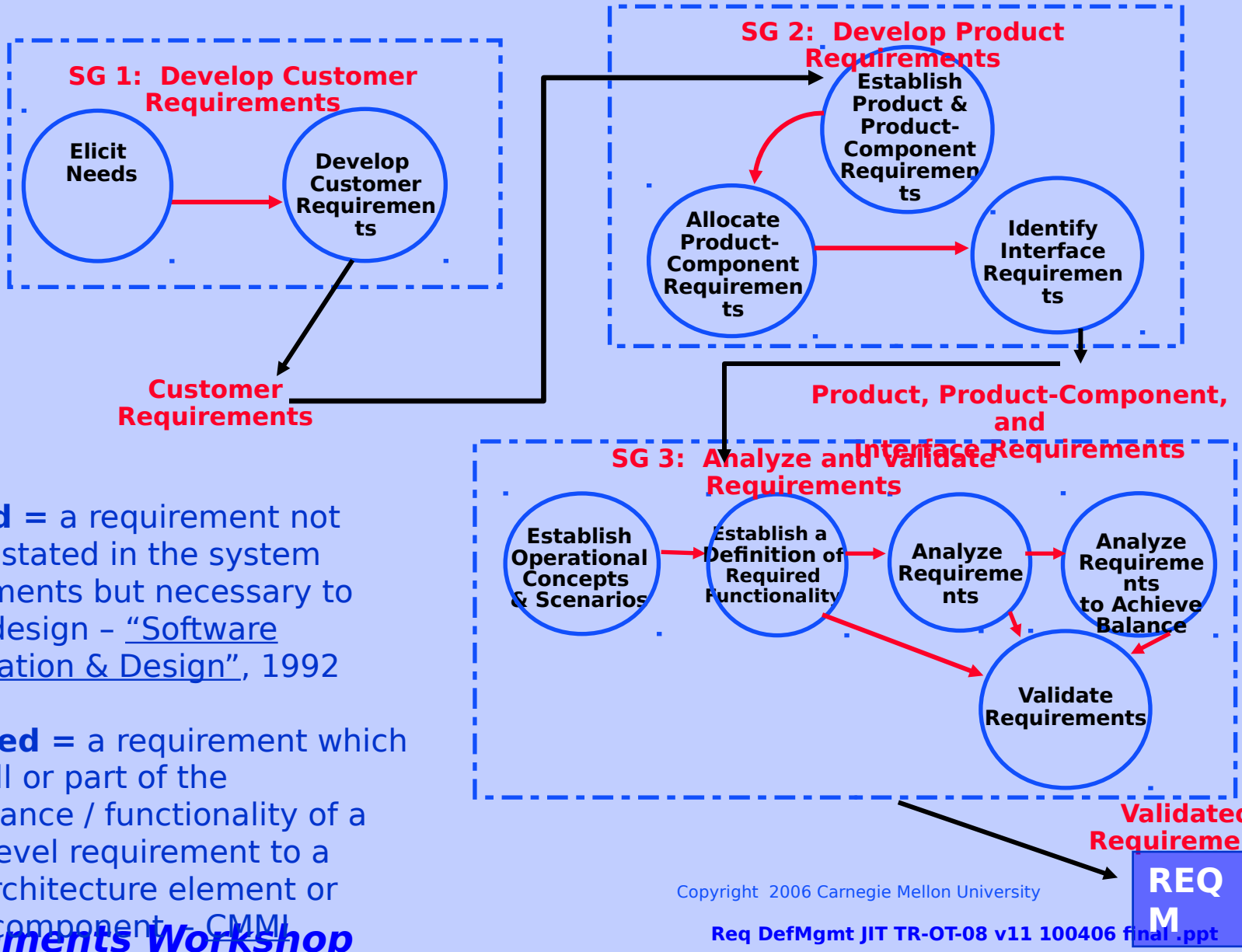
Requirements Management


Manage Requirements
Obtain an Understanding of Requirements
Requirements
Identify Inconsistencies Between Project Work and Reqmts
Obtain Commitment to Requirements
Manage Requirements Changes
Maintain Bidirectional Traceability of Requirements
Traceability Matrix or Requirements Tracking System

Copyright 2006 Carnegie Mellon University

SPAWAR Systems Center San Diego

Requirements Development - SGs



Derived = a requirement not directly stated in the system requirements but necessary to satisfy design – "Software Specification & Design", 1992

Allocated = a requirement which levies all or part of the performance / functionality of a higher level requirement to a lower architecture element or design component – CMMI Glossary

Copyright 2006 Carnegie Mellon University